

(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平10-304372

(43)公開日 平成10年(1998)11月13日

(51)Int.Cl.
H 0 4 N 7/32

識別記号

F I
H 0 4 N 7/137

Z

審査請求 未請求 請求項の数7 O.L (全 18 頁)

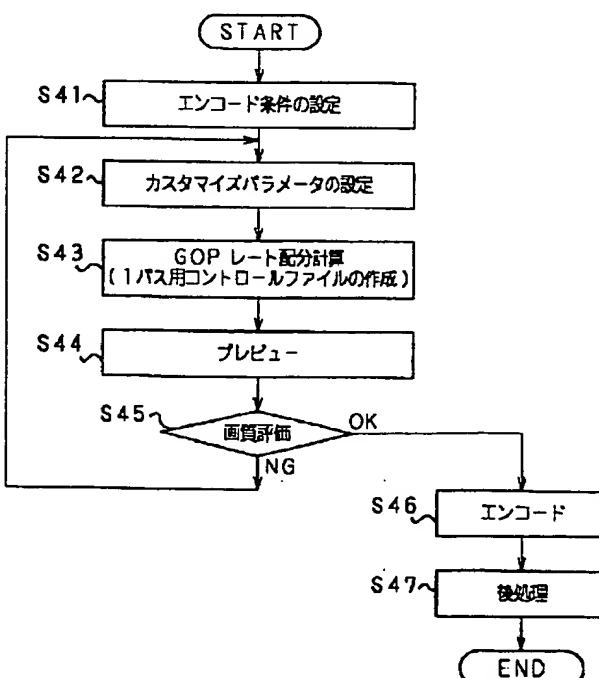
(21)出願番号	特願平9-113144	(71)出願人	000002185 ソニー株式会社 東京都品川区北品川6丁目7番35号
(22)出願日	平成9年(1997)4月30日	(72)発明者	五十崎 正明 東京都品川区北品川6丁目7番35号 ソニ ー株式会社内

(54)【発明の名称】 画像符号化方法および装置、画像伝送方法

(57)【要約】

【課題】 壓縮符号化処理時間を短縮すると共に、符号化ビットレートを任意に配できる画像符号化方法および装置、画像伝送方法を提供する。

【解決手段】 ステップS 4 1で設定されたエンコード条件に従って、ステップS 4 2でエンコード素材（ビデオ素材）に対するビット配分の重み係数を含むカスタマイズパラメータを設定する。ステップS 4 3で、このパラメータに基づいてGOPに対する符号化レートの配分を計算し、コントロールファイルを作成する。このとき、仮想的な符号化難易度を導入することにより、GOP単位でカスタマイズを行うことができる。その後、必要に応じてステップS 4 4でプレビューを行い、ステップS 4 5で画質評価した後に、ステップS 4 3で作成されたコントロールファイルを用いて1パス画像符号化を実行し、ステップS 4 7で後処理を行って画像符号化処理を終了する。



【特許請求の範囲】

【請求項1】 ビデオ素材に対してビット配分の重み係数を含むカスタマイズパラメータを設定するカスタマイズパラメータ設定工程と、

上記設定されたカスタマイズパラメータに基づいて、少なくとも1つのフレーム内符号化画像と、フレーム間順方向予測符号化画像および双方向予測符号化画像を含んで構成される画像符号化グループを単位とする符号化ビットレートを、それらの総データ量が予め設定された所定のデータ量を越えないように配分するレート配分工程とを有することを特徴とする画像符号化方法。

【請求項2】 上記カスタマイズパラメータ設定工程において、ビデオ素材に対して画質調整を行うカスタマイズ範囲を指定することを特徴とする請求項1記載の画像符号化方法。

【請求項3】 上記レート配分工程において、上記画像符号化グループを構成する画像毎に設定される仮想符号化難易度と上記重み係数を用いて、符号化ビットレートを配分することを特徴とする請求項1記載の画像符号化方法。

【請求項4】 ビデオ素材に対してビット配分の重み係数を含むカスタマイズパラメータを設定するためのカスタマイズパラメータ設定手段と、

上記設定されたカスタマイズパラメータに基づいて、少なくとも1つのフレーム内符号化画像と、フレーム間順方向予測符号化画像および双方向予測符号化画像を含んで構成される画像符号化グループを単位とする符号化ビットレートを、それらの総データ量が予め設定された所定のデータ量を越えないように配分するための制御手段とを備えることを特徴とする画像符号化装置。

【請求項5】 上記カスタマイズパラメータ設定手段は、ビデオ素材に対して画質調整を行うカスタマイズ範囲を指定することを特徴とする請求項4記載の画像符号化装置。

【請求項6】 上記制御手段は、上記画像符号化グループを構成する画像毎に設定される仮想符号化難易度と上記重み係数を用いて、符号化ビットレートを配分することを特徴とする請求項4記載の画像符号化装置。

【請求項7】 ビデオ素材に対してビット配分の重み係数を含むカスタマイズパラメータを設定するカスタマイズパラメータ設定工程と、

上記設定されたカスタマイズパラメータに基づいて、少なくとも1つのフレーム内符号化画像と、フレーム間順方向予測符号化画像および双方向予測符号化画像を含んで構成される画像符号化グループを単位とする符号化ビットレートを、それらの総データ量が予め設定された所定の伝送データ量を越えないように配分するレート配分工程と、

上記ビットレートが配分された符号化画像を伝送する伝送工程とを有することを特徴とする画像伝送方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、少なくとも1つのフレーム内符号化画像と、フレーム間順方向予測符号化画像および双方向予測符号化画像を含んで構成される画像符号化グループを単位として画像を圧縮符号化する画像符号化方法および装置、ならびに上記の圧縮符号化された画像を伝送する画像伝送方法に関する。

【0002】

【従来の技術】 デジタルビデオディスク (Digital Video Disk : DVD) やビデオCDなどの、いわゆるパッケージメディアに蓄積されるビデオデータには、圧縮符号化処理が施されている。このような圧縮符号化処理 (エンコード) を行う際には、ビデオ素材を画像符号化装置 (エンコーダ) に2回読み込んで処理する2バス符号化 (2バスエンコード) と呼ばれる方法が用いられる。

【0003】 この2バスエンコードでは、1バス目の読み込み時に、エンコードされる画像毎の符号化難易度 (Difficulty) がビデオ素材の全体に亘って測定され、その符号化難易度に基づいて、データ圧縮されたビデオ素材がパッケージメディアの記録容量に収まるように符号化の際のビット配分 (Bit assign) が計算される。そして、2バス目の読み込み時には、上記のビット配分に従って、ビデオ素材が可変レートで圧縮符号化される。

【0004】 以上のような可変レートの2バスエンコードは、限られた符号化ビット量を有効に利用できるため、圧縮されたビデオ素材をデジタルビデオディスク (Digital Video Disk : DVD) やビデオCDのような所定の記録容量のパッケージメディアに蓄積する際に有効である。

【0005】 しかし、2バスエンコードを行うためには、ビデオ素材の長さの少なくとも2倍の作業時間が必要であるため、制作コストが上昇する一因になる。このため、画質よりも作業時間の短縮を優先させる場合には、1バスの固定レートのエンコードが選択されることがある。

【0006】

【発明が解決しようとする課題】 上記の1バスエンコードによれば、ビデオ素材を1回だけエンコーダに読み込まれよいため、2バスエンコードに比べて作業時間を短縮できる。しかし、固定レートでエンコードするため、複数のエンコード単位からなるビデオ素材のうちの特定のエンコード単位の符号化ビットレートだけを他のエンコード単位よりも上げる等の処理ができなかった。このような要求があるビデオ素材としては、例えば、広告用のビデオクリップなどがあるために問題であった。

【0007】 また、エンコードを行う前に、ビデオ素材の静止画の部分や、被写体の動きが特に激しく複雑な画像の部分について、タイムコードなどの時間情報が分か

っている場合には、符号化が特に難しいと予測される部分についてのみ符号化ビットレートを上げたいという要求がある。さらに、1バスエンコードを行った結果の画質に問題があるときには、特定の範囲のレートだけを調整したいという要求も生じる。しかし、固定レートのエンコードは、このような要求には応じられないという問題があった。

【0008】本発明は、このような問題を解決するために行われたものであり、エンコード作業時間を短縮することができる1バスのエンコードにおいて、ビデオ素材に配分される符号化ビットレートを可変して圧縮符号化できる画像符号化方法および装置、ならびに上記の圧縮符号化を施して画像を伝送する画像伝送方法を提供することを目的としている。

【0009】

【課題を解決するための手段】上記の課題を解決するために提案する本発明の画像符号化方法は、ビデオ素材に対してビット配分の重み係数を含むカスタマイズパラメータを設定するカスタマイズパラメータ設定工程と、上記設定されたカスタマイズパラメータに基づいて、少なくとも1つのフレーム内符号化画像と、フレーム間順方向予測符号化画像および双方向予測符号化画像を含んで構成される画像符号化グループを単位とする符号化ビットレートを、それらの総データ量が予め設定された所定のデータ量を越えないように配分するレート配分工程とを有することを特徴とするものである。

【0010】また、上記の課題を解決するために提案する本発明の画像符号化装置は、ビデオ素材に対してビット配分の重み係数を含むカスタマイズパラメータを設定するためのカスタマイズパラメータ設定手段と、上記設定されたカスタマイズパラメータに基づいて、少なくとも1つのフレーム内符号化画像と、フレーム間順方向予測符号化画像および双方向予測符号化画像を含んで構成される画像符号化グループを単位とする符号化ビットレートを、それらの総データ量が予め設定された所定のデータ量を越えないように配分するための制御手段とを備えることを特徴とするものである。

【0011】また、上記の課題を解決するために提案する本発明の画像伝送方法は、ビデオ素材に対してビット配分の重み係数を含むカスタマイズパラメータを設定するカスタマイズパラメータ設定工程と、上記設定されたカスタマイズパラメータに基づいて、少なくとも1つのフレーム内符号化画像と、フレーム間順方向予測符号化画像および双方向予測符号化画像を含んで構成される画像符号化グループを単位とする符号化ビットレートを、それらの総データ量が予め設定された所定の伝送データ量を越えないように配分するレート配分工程と、上記ビットレートが配分された符号化画像を伝送する伝送工程とを有することを特徴とするものである。

【0012】上記の本発明によれば、ビデオ素材のエン

コード作業時間を短縮することができる1バスのエンコードにおいて、ビデオ素材の任意の部分に仮想的な符号化難易度に基づいて重み付けされた符号化ビットレートを配分できるようにしたため、高画質を保ちながらデータ量を効果的に圧縮できる画像符号化方法および装置、ならびに上記の圧縮符号化を施して画像を伝送する画像伝送方法を提供できる。

【0013】

【発明の実施の形態】次に、本発明の好ましい実施の形態について図面を参照しながら説明する。

【0014】以下では、まず本発明の画像符号化装置の構成について説明する。次に、上記の画像符号化装置の構成に基づいて、本発明の画像符号化方法の基本となる2バスエンコードおよび固定レートの1バスエンコードについて説明し、その後、本発明の画像符号化方法について説明する。なお、本発明の画像伝送方法については最後に説明する。

【0015】図1は、本発明の画像符号化装置の実施の一形態であるビデオエンコードシステムの構成例を示すブロック図である。このビデオエンコードシステムは、ビデオ素材を圧縮符号化して、DVDなどの記録媒体(パッケージメディア)に蓄積するためのものである。

【0016】スーパーバイザコントローラ1は、オペレーティングシステムを構成するプログラムのうち、このビデオエンコードシステム全体の動作を監視して効率的に制御するプログラムであるスーパーバイザを実行するコントローラである。

【0017】このビデオエンコードシステムでは、スーパーバイザコントローラ1が、DVDのオーサリングシステム全体の管理を行い、ビデオ、オーディオ、字幕やメニューなどのデータをエンコードするための各エンコードシステムにエンコード条件を与えたり、エンコード結果の報告を受ける。

【0018】なお、オーサリングとは、映像、音声、字幕などのデータを、それぞれのデータ圧縮方式でエンコードして、最終的にDVDなどの記録媒体に記録できるデジタルデータにまとめることをいう。

【0019】ビデオエンコードコントローラ20は、ビデオエンコードを制御するシステム全体を管理するためのものであり、ネットワーク2を介してスーパーバイザコントローラ1に接続されている。

【0020】このビデオエンコードシステムでは、スーパーバイザコントローラ1から送られる「v.enc」というファイルによって、ビデオエンコード条件が指定される。そして、ビデオエンコードコントローラ20側からは、RAID (Redundant Arrays of Inexpensive Disk es) 16上のアドレス「v.adr」と、データ「vxxxx.au[i]」がスーパーバイザコントローラ1に報告される。上記の「v.adr」は、エンコード結果のビット列(ビットストリーム)が書き込まれるRAID16のアドレスで

ある。「vxxx.aui」は、エンコード結果のビットストリームがオーディオや字幕、メニュー等のサブピクチャと多重化（マルチプレックス）される際に必要とされるデータである。なお、RAID16は、複数台のハードディスクドライブ（HDD）を並列接続して冗長度を付与することにより、記録容量や転送速度を向上させた記録再生装置である。

【0021】ビデオエンコードコントローラ20は、ビデオ素材に対して符号化ビットレートを配分（以下では単にビット配分ともいう。）する際の重み係数を含むカスタマイズパラメータを設定するためのカスタマイズパラメータ設定手段であるグラフィカルユーザインターフェース（GUI）21と、ビット配分計算部22と、MPEGエンコーダコントローラ23と、デジタルVTR（DVTR）コントローラ24を備えており、パーソナルコンピュータなどを用いて構成される。

【0022】ユーザは、GUI21を介して、ビット配分計算部22に格納されているプログラム「BIT_ASSIGN」と、MPEGエンコーダコントローラ23の3つのプログラムを管理することができる。また、ユーザは、GUI21を介してDVTRコントローラ24を管理することもできる。

【0023】MPEGエンコーダコントローラ23は、フレーム間順方向予測符号化画像および双方方向予測符号化画像を含んで構成される画像符号化グループを単位とする符号化ビットレートを、それらの総データ量が予め設定された所定のデータ量を越えないように配分するための制御手段であり、後述するMPEGエンコーダ25を制御する。

【0024】また、DVTRコントローラ24は、DVTR（デジタルVTR）17を制御するためのものである。このDVTR17は、MPEGエンコーダ25と接続されている。

【0025】MPEGエンコーダ25は、ビデオデータなどをMPEG（Moving Picture Experts Group）方式により圧縮符号化するためのものである。ここでは、ビデオデータの時間方向の冗長度が、動き補償予測により除去される。なお、動き補償予測は、前画像から現在の画像を予測して符号化する予測符号化に際して、画像中の動きがある部分の変化分（動きのずれの量）を表す「動きベクトル」を符号化することによりビデオデータを効果的に圧縮する方法である。

【0026】また、MPEGエンコーダ25は、エンコードした結果を表示するためのモニタ18に接続され、また、エンコード結果のビットストリームが記録されるRAID16にも接続されている。

【0027】MPEGエンコーダ25では、前画像を用いる予測符号化を行わずにフレーム内だけで符号化される画像であるIピクチャ（Intra-Picture）、過去の画像から現在の画像を予測して符号化される画像であるP

ピクチャ（Predictive-Picture）、過去と未来の両方向の画像から現在の画像を予測して符号化される画像であるBピクチャ（Bidirectionally Predictive-Picture）が生成される。

【0028】そして、生成されたピクチャは、少なくとも1枚のIピクチャを必ず含む画像符号化グループであるGOP（Group of Pictures）として構成される。

【0029】図2は、GOP内の各ピクチャの配置例を示している。ここでは、1つのGOPを構成するピクチャ（フレーム）枚数Nが15であり、IピクチャまたはPピクチャが現れる周期Mが3である場合を例示している。

【0030】Iピクチャは、GOPの独立性を保ち、ランダムアクセスを可能にするために必要なピクチャであり、通常は1フィールドに1枚（1フレームに2枚）の割合で含まれている。そして、15枚（0.5秒）程度の画面が1つのGOPとされて取り扱われるのが通常である。

【0031】表示順におけるGOPの先頭は、Iピクチャの前にある最初のPピクチャ、またはIピクチャの次にあるBピクチャである。また、GOPの最後は、次のIピクチャの前にある最初のPピクチャである。

【0032】次に、ビデオ素材を圧縮符号化処理するための2パスエンコードについて説明する。

【0033】図3は、図1のビデオエンコードシステムにおいて、2パスエンコードを行う際の処理手順を示すフローチャートである。

【0034】まず、ステップS71で、エンコード条件を示すファイル「v.enc」が、スーパーバイザコントローラ1からネットワーク2を経由してビデオエンコードコントローラ20に与えられる。そして、MPEGエンコーダコントローラ23は、このファイル「v.enc」により示される、ビデオ素材に割り当てられる符号化のためのビット総量や最大レートなどのエンコード条件を設定する。

【0035】ステップS72では、MPEGエンコーダコントローラ23が、エンコード素材（ビデオ素材）の符号化難易（Difficulty）を測定する。この符号化難易度については後述する。また、ここでは、上記のビデオ素材の各画素のDC（直流成分）値や動きベクトル量MEなども測定される。そして、これらの測定結果に基づいてファイルが作成される。

【0036】上記の符号化難易度（Difficulty）は、次のように測定される。エンコード素材となるビデオデータは、例えばDVTR（デジタルVTR）17により磁気テープにマスターとして収録されている。このマスターは、DVTRコントローラ24により制御されながらDVTR17で再生されて、MPEGコントローラ23でピクチャ毎の符号化難易度（Difficulty）が測定される。

【0037】このとき、符号化時の量子化ステップ数を固定値に設定しておき、エンコード素材を符号化する際の発生ビット量が符号化難易度として測定される。この測定により、被写体の動きが多く、高い周波数成分を多く含む画像は、符号化時の発生ビット量が大きくなるため符号化難易度が高いとされる。一方、静止画や平坦な部分が多い画像は、発生ビット量が少なくなるため符号化難易度が低いとされる。

【0038】次に、ステップS73では、ビデオエンコードコントローラ20が、ステップS71で設定されたエンコード条件に基づいて、割り当てビット量（ターゲット量）の配分計算を行う。このビットの配分計算は、ステップS72で測定された各ピクチャの符号化難易度に応じて、ビット配分計算部22に格納された計算プログラム「BIT_ASSIGN」を実行することにより行われるものである。なお、このビットの配分計算については後述する。

【0039】そして、ステップS74では、ステップS73で得たビット配分計算の結果を用いてエンコードされた画像の画質を確認するためのプレビュー（Preview）が行われる。このプレビューは、MPEGエンコーダ25に内蔵されているローカルデコーダを用いるモードであり、エンコード結果のビットストリームの任意の範囲を指定することができる。なお、上記のエンコード結果は、RAID16には転送されない。

【0040】ステップS75では、ユーザーが、プレビュー画像の画質を確認して、上記のビット配分計算の結果を用いて最終的にエンコードを実行するかどうかを判断する。この評価で画質に問題がないとされる場合にはステップS78に進む。一方、ステップS75の画質評価で問題があるとされる場合には、ステップS76に進む。

【0041】ステップS76では、プレビューされたビットストリームのうちの画質に問題があるとされた部分に対して、ビット配分を多く（ビットレートを高く）したり、ビデオデータのR、G、Bの各成分に適切なノイズリダクション処理を行う際のフィルターレベルを調整するなどの、画質調整のためのカスタマイズが行われる。

【0042】そして、ステップS77でビット配分の再計算を行った後にステップS74に戻り、カスタマイズを行った部分のプレビューを行う。そして、画質に問題がないとされればステップS78に進む。

【0043】ステップS78では、MPEGエンコーダ25が、ステップS77で計算されたビット配分に従って、エンコード素材全体のエンコードを実行する。

【0044】そして、ステップS79で、エンコード結果のビットストリームが、SCSI（Small Computer System Interface）を介してRAID16に直接転送されて書き込まれる。また、ビデオエンコードコントロー

ラ20は、ステップS78におけるエンコードの終了後に、エンコード結果情報をネットワーク2を経由してスーパーバイザコントローラ1に報告する。このような後処理が終了すると、2バスエンコード処理が終了したことになる。

【0045】なお、この図3に示す処理手順において、ステップS72、ステップS74およびステップS78はオンライン処理であり、その他の各ステップの処理はオフラインで行われる。

【0046】次に、図3に示した処理手順のステップS73で行われるビットの配分計算について詳細に説明する。

【0047】まず、エンコード結果が最終的に蓄積されるDVDなどの記録媒体の容量のうちの、ビデオデータに割り当てられるビット総量「QTY_BYTES」と最大ビットレート「MAXRATE」が、スーパーバイザコントローラ1から指定される。

【0048】これに対して、ビット配分計算部22は、最大ビットレート「MAXRATE」以下になるように制限を加えた総ビット量「USB_BYTES」を求め、この値からGOPのヘッダ（GOP header）として必要なビット量「TOTAL_HEADER」を差し引いた値と、全体のフレーム総数からターゲット数の総和の目標値となる「SUPPLY_BYTES」を算出する。そして、エンコード結果が「SUPPLY_BYTES」の大きさに収まるように、各ピクチャに対する符号化ビット量（以下では、ターゲット（target）量という。）を配分する。

【0049】なお、計算プログラム「BIT_ASSIGN」は、例えばビデオエンコードコントローラ20のメモリに格納されており、上記のビットの配分計算は、実際にはMPEGエンコーダコントローラ23で実行される。

【0050】図4は、上記のビットの配分計算の処理手順を示すフローチャートである。

【0051】ここでは、エンコード素材に対して、まずビット量がGOP単位で配分され、その後、各GOP内で各ピクチャの符号化難易度（Difficulty）に応じてビット配分が行われるものとして説明する。このとき、各GOP毎の符号化難易度の和である「GOP_DIFF」に応じて、エンコードする際のGOP単位のビット割り当て量「GOP_TARGET」が配分される。

【0052】まず、ステップS81で、エンコード条件が入力される。このエンコード条件は、前述したように、ビット総量「QTY_BYTES」と最大ビットレート「MAXRATE」などであり、スーパーバイザコントローラ1から送られる。

【0053】次に、ステップS82で、図3のステップS72における符号化難易度（Difficulty）の測定に基づいて作成されたファイルがそのまま読み込まれる。

【0054】ステップS83では、上記の符号化難易度と併せて測定された各画像のDC（直流成分）値や動き

ベクトル量M Eの大きさのパラメータの変化量から、シーンが変化するシーンチェンジ点が検出され、後述するシーンチェンジ処理が行われる。

【0055】なお、ステップS 8 3におけるシーンチェンジ検出／処理は、本発明の出願人が特願平8-274094号明細書および図面にて既に開示した「映像信号処理装置」に応じて行われる処理である。この「映像信号処理装置」は、各フレームの映像信号の直流成分のレベルを検出し、この直流レベルを曲線近似して得られる誤差値を用いて上記映像信号のシーンチェンジのフレームを検出するものである。

【0056】図5は、シーンチェンジ指定ポイントで、PピクチャがIピクチャに変更される様子を示している。

【0057】このように、シーンチェンジが検出されたポイントでは、Pピクチャを、前画像を用いる予測符号化を行わずにフレーム内だけで符号化される画像であるIピクチャに変更することにより、画質改善が図られる。

【0058】次に、ステップS 8 4で、チャプタ(CHAPTER)境界処理が行われる。DVD再生装置におけるチャプタサーチ時には、表示される画像が、特定されない画像(ピクチャ)からチャプタ指定ポイントにジャンプしていくことになる。その場合にも表示される再生画像が乱れないようにするために、チャプタ指定ポイントが必ずGOPの先頭になるようにピクチャタイプが変更される。

【0059】図6は、このようなチャプタ指定ポイントで、PピクチャがIピクチャに変更される様子を示している。

【0060】そして、ステップS 8 5では、符号化難易度(Difficulty)の値が変更後のピクチャタイプに合わせた値に補間／補正される。これは、ステップS 8 2およびステップS 8 3における一連の処理の結果、ピクチャタイプ(I, P, Bピクチャ)が変更されたことによ

USB_BYTES

= min(QTY_BYTES, MAXRATE × KT × total_frame_number) (1)

ここで、ビデオ信号規格がNTSC方式である場合には
KT = 1/8 (bits) / 30 (Hz) であり、PAL方式の場合は1/8 (bits) / 25 (Hz) である。

【0070】「total_frame_number」はエンコードするビデオ素材のフレーム総数であり、「min(s, t)」はs, tのうちの小さい方を選択する関数である。

SUPPLY_BYT = (USB_BYTES) - (TOTAL_HEADER) (2)

【0073】

また、全てのピクチャの符号化難易度(Difficulty)の総和「DIFFICULTY_SUM」は、(3)式で表される。

DIFFICULTY_SUM = Σ (Difficulty)

さらに、GOP単位のピット割り当て量「GOP_TARGET」の最小値を(4)式のようにおくと(5)式を得る。

B = GOP_MINBYTES

り符号化難易度測定時のピクチャタイプも変更されるためである。

【0061】ステップS 8 6では、ステップS 8 5で補間／補正されて得られた符号化難易度と、エンコード素材全体に対して与えられたビット(ターゲット)数の総和の目標値である「SUPPLY_BYTES」に応じて、各ピクチャ毎のターゲットビット量が計算される。

【0062】そして、ステップS 8 7で、エンコード結果のビットストリームをRAID16に書き込む際のアドレスが計算される。

【0063】そして、ステップS 8 8では、エシコーダ用のコントロールファイルが作成される。

【0064】以上の手順により、図3のステップS 7 3におけるビット配分計算が行われる。

【0065】ところで、上記のビット配分計算において、各GOP毎の符号化難易度の和である「GOP_DIFF」と、エンコードする際のGOP単位のピット割り当て量「GOP_TARGET」とを変換するために、適当な評価関数が用いられている。

【0066】図7は、上記の「GOP_DIFF」と「GOP_TARGET」とを変換するための最も簡単な関数の例を示している。この例では、縦軸Yを「GOP_TARGET」、横軸Xを「GOP_DIFF」として、Y=AX+Bという形の評価関数を用いている。なお、全てのピクチャの符号化難易度の総和「DIFFICULTY_SUM」は、予め算出されているものとしている。

【0067】次に、上記の評価関数を用いて行われるビット配分計算について具体的に説明する。

【0068】まず、最大ピットレート「MAXRATE」以下になるように制限を加えた総ビット量「USB_BYTES」を、スーパーバイザコントローラ1から与えられたビット総量「QTY_BYTES」と最大ピットレート「MAXRATE」を用いて(1)式のように求める。

【0069】

ES = (1)式で求めた制限を加えた総ビット量「USB_BYTES」からGOPのヘッダに必要なビット量「TOTAL_HEADER」を差し引いて(2)式のように求める。

【0072】

【0071】ターゲット数の総和の目標値「SUPPLY_BYT

ES」は、(1)式で求めた制限を加えた総ビット量「US

B_BYTES」からGOPのヘッダに必要なビット量「TOT

L_HEADER」を差し引いて(2)式のように求める。

【0072】

$$\Sigma y = A \times \Sigma x + B \times n$$

ここで、 $\Sigma y = SUPPLY_BYTES$ 、 $\Sigma x = DIFFICULTY_SU$

M であり、nはGOPの総数である。これより(6)式

$$A = (SUPPLY_BYTES - B \times n) / (DIFFICULTY_SUM) \quad (6)$$

従って、各GOP毎のターゲット量は(7)式で表される。

$$GOP_TARGET = A \times (GOP_DIFF) + B \quad (7)$$

その後、各GOP内で、ピクチャ毎の符号化難易度(Difficulty)に応じたビット配分が行われる。GOP内の各ピクチャに配分される符号化ビット量を符号化難易度

$$target(k) = (GOP_TARGET \times difficulty(k)) / (GOP_DIFF) \quad (1 \leq k \leq GOP\text{内のピクチャ数}) \quad (8)$$

この場合には、ビデオ素材の中に符号化が極端に難しい(すなわち「GOP_DIFF」が大きい)ピクチャがあると、「GOP_TARGET」が非常に大きい量になってしまい、許容されている最大ビットレートを越えてしまうことがある。このため、最大のターゲット量が、「GOP_MAXBYTE_S」などの固定量を用いて制限されるようにする必要がある。同様に、最小のターゲット量も固定値「GOP_MINBYTES」で制限される。このようなターゲット量の制限は、具体的には、以下に説明するアルゴリズムにより行われる。

$$OCCUPANCY_UP(0) = VBVMAX * 2/3 \quad (9)$$

なお、以下の説明では、「OCCUPANCY_UP」がグラフ上の各ピクチャの上側のポイントを意味し、「OCCUPANCY_DOWN」がグラフ上の各ピクチャの下側のポイントを意味するものとする。

【0081】例えば、DVDのバッファサイズ「VBVMAX (1.75Mbps)」に対して、k番目のピクチャのバッファのスタート点を「Occupancy_up(k)」とし、そのピクチャのターゲット量を「target(k)」とすると、ビ

$$OCCUPANCY_DOWN(k) = OCCUPANCY_UP(k) - target(k) \quad (10)$$

$$OCCUPANCY_UP(k+1) = OCCUPANCY_DOWN(k) + SYSTEM_SUPPLY \quad (11)$$

(11)式で表されるビット配分後のバッファ内のデータ残量は、図8中の右上に向かって上昇する矢印で表される量に相当する。すなわち、各ピクチャに配分されるビットレートが大きいほど、この傾きは大きくなり、データがバッファに蓄積されやすいことを示す。なお、バッファが一杯になった場合には、デコーダからバッファへのデータ供給が中止されるため、バッファのオーバーフローを考慮する必要はない。このことは、バッファ残量を、ある設定値ちょうどになるように制御する必要はなく、設定値以上になるように制御すれば良いことを意

$$SYSTEM_SUPPLY = MAXRATE(bps) * KT \quad (12)$$

図9は、GOP単位のターゲットビット配分計算の結果の一例を示している。ここで、図9(A)は、前述した評価関数と、最大ビットレート「GOP_MAXRATE」による制限を考慮して求めたターゲット量に対して、上記のVBVバッファ計算を行った結果の一例である。この図中

$$\Sigma y = A \times \Sigma x + B \times n \quad (5)$$

を得る。

【0075】

$$A = (SUPPLY_BYTES - B \times n) / (DIFFICULTY_SUM) \quad (6)$$

【0076】

の大きさに比例させる場合には、各ピクチャのターゲット量は(8)式で求められる。

【0077】

$$target(k) = (GOP_TARGET \times difficulty(k)) / (GOP_DIFF) \quad (1 \leq k \leq GOP\text{内のピクチャ数}) \quad (8)$$

【0078】MPEG(Moving Picture Experts Group)によるビデオエンコード時には、仮想デコーダのバッファ残量を考慮しながらビット配分することが義務付けられている。この仮想バッファ残量の計算はVBV(Video buffering verifier)と呼ばれる。

【0079】図8は、このVBVの計算手順を例示している。計算の最初の「OCCUPANCY_UP(0)」は、次の(9)式に示すように固定値(この例では「VBVMAX * 2/3」)からスタートする。

【0080】

$$OCCUPANCY_UP(0) = VBVMAX * 2/3 \quad (9)$$

クチャにビットを配分した後のバッファ内のデータ残量「OCCUPANCY_DOWN(k)」は(10)式で表される。

【0082】このバッファには、ビデオ素材のデータ量に応じたビットレートとされるデータ量「SYSTEM_SUPPLY」が、デコーダから供給され蓄積される。また、ピクチャにビットを配分した後のバッファ内のデータ残量「OCCUPANCY_UP(k+1)」は(11)式で表される。

【0083】

味している。逆に、各ピクチャに配分されるデータ量が大きすぎると、バッファに蓄積されるデータが減少してしまうことになる。

【0084】つまり、このバッファ内のデータ残量が一定値以下にならないように、ターゲットビット量が計算される。そして、ビデオ素材のデータ量に応じたビットレートとされるデータ量「SYSTEM_SUPPLY」が(12)式で求められる。

【0085】

の1番目、4番目、7番目のピクチャでは、VBVの下限である「VBVMIN」の値を下回っている。そこで、VBVが「VBVMIN」を下回ったピクチャを含むGOPのターゲット量が削減される。

【0086】GOP内で、VBV制限を加える前のタ

ゲット量を用いてVBV計算を実行したときの「OCCUPANCY」の最小値を「Occ_min」とすると、VBVが制限された値を越えないようにするための調整量は以下の式で表される。ここで、VBV制限を開始するスタート点

$$r = (Occ_{start} - VBVMIN) / (VBVSTART - Occ_{min}) \quad (13)$$

ただし、各ターゲットに対して(14)式が成り立つとする。

$$target(j) = target(j) \times r \quad (kstart \leq j \leq k) \quad (14)$$

また、図9(B)は、上記のVBV制限を行った後の計算結果の例を示している。このように求められたターゲット量に基づいて作成されたコントロールファイルを用いてエンコード処理を行うことにより、ビデオ素材の画像毎の符号化難易度に応じてビットレートを配分する可変ビットレートエンコードが実行される。

【0088】しかし、2バスエンコードを行うためには、前述したように、ビデオ素材の長さの少なくとも2倍の作業時間が必要である。このため、制作コストが上昇するという問題があり、画質よりも作業時間の短縮を優先させる場合には、以下に説明する1バスの固定レートのエンコードが選択される。

【0089】次に、従来行われている固定レートの1バスエンコードについて、再び図1のビデオエンコードシステムの構成を参照しながら説明する。

【0090】図10は、固定レートの1バスエンコードにおける処理手順を示すフローチャートである。

【0091】まず、ステップS21では、スーパーバイザコントローラ1から、ネットワーク2を経由してエンコード条件「v.enc」が与えられる。この「v.enc」は、ビデオ素材に割り当てる符号化ビットの総量や最大レートなどに関する情報である。そして、MPEGエンコーダコントローラ13は、このエンコード条件に基づいてMPEGエンコーダ25を設定する。

【0092】ステップS22では、1バスエンコード用のコントロールファイルが作成される。このコントロールファイルの具体的な作成手順については後述する。

【0093】ステップS23では、ステップS22で作成されたコントロールファイルを用いてエンコード処理が行われる。

【0094】最後に、ステップS24で、エンコード結果のビットストリームをRAID16に書き込むためのアドレスの設定や、エンコーダ用コントロールファイル

```
gop_cycle[15]
= {"B", "B", "I", "B", "B", "P", "B", "B", "P", "B", "B", "P", "B", "B", "B", "P"}
p_type[k] = gop_cycle [(k+2) mod (gop_n)]
```

(15)

次に、ステップS33では、CELL境界(チャプタ境界)処理が行われる。この処理は、図6に示した2バスエンコードにおけるチャプタ境界処理と同様であるので、ここでは説明を省略する。

【0101】次に、ステップS34で、平均のビットレ

「kstart」は、「OCCUPANCY_UP(k)」が基準値「VBVLINE(例えばVBVMAX * 3/4)」以上のkの値であり、このときの「OCCUPANCY_UP(k)」の値を「Occ_start」とする。OCCUPANCY_MIN < VBVMINのときには

$$(VBVSTART - Occ_{min}) \quad (13)$$

【0087】

$$target(j) = target(j) \times r \quad (kstart \leq j \leq k) \quad (14)$$

を出力する等の後処理が行われ、1バスエンコード処理が終了する。

【0095】なお、ステップS23のエンコード工程を除く各ステップにおける処理は、オフラインで行われる処理である。

【0096】ところで、上記のようなエンコードにおいて、エンコード素材(ビデオ素材)を1本の磁気テープに記録することができない場合には、複数の磁気テープに分割して記録されることになる。この場合には、デジタルVTR17で磁気テープの入れ替え作業を行う必要があるため、エンコード処理を連続して行うことができない。このような場合に、処理を中断することなく一度にエンコードできる素材のエンコード単位(ENCU:Encode Unit)と定義する。図11は、前述した1バスエンコード用コントロールファイルの作成手順を示すフローチャートである。

【0097】まず、ステップS31で、エンコード条件が入力される。ここでは、前述した2バスエンコードの場合と同様に、ディスク容量のうちのビデオに割り当てられたビット総量「QTY_BYTES」と、最大ビットレート「MAXRATE」が、オーサリングシステムから指定される。そして、これに対して、最大ビットレート以下になるよう制限を加えられた総ビット量「USB_BYTES」が求められる。

【0098】ステップS32では、ピクチャタイプが指定される。ここでは、ピクチャタイプが、先頭のフレームから順にI, B, B, P, B, B, ...と規則正しく設定されていく。

【0099】各エンコードユニット(ENCU)の先頭からk番目のフレーム番号kのピクチャタイプを「p_type[k]」とし、GOPの長さを「gop_n」とすると、「p_type[k]」は(15)式で与えられる。

【0100】

ートが計算される。

【0102】固定レートの1バスエンコードにおける平均ビットレート「CONSTANT_RATE」は、以下の(17)式で表される。

【0103】

$$\begin{aligned} \text{USB_BYTES} \\ = \min (\text{QTY_BYTES} - \text{MAXRATE} \times \text{KT} \times \text{total_frame_number}) \end{aligned} \quad (16)$$

$$\begin{aligned} \text{CONSTANT_RATE} \\ = \text{USB_BYTES} / \text{total_frame_number} / \text{KT} \end{aligned} \quad (17)$$

なお、上記の(16)式は、前述した2バスエンコードにおける(1)式と同様の式であり、NTSC方式の場合には $KT = 1/8$ (bits) / 30 (Hz) であり、PAL方式の場合には $1/8$ (bits) / 25 (Hz) である。

【0104】また、「total_frame_number」はエンコード素材(ビデオ素材)のフレーム総数であり、「min (s, t)」はs, tのうちの小さい方を選択する関数である。

【0105】ステップS35では、GOP毎に配分されるビットレートが設定される。このビットレートの平均がエンコード時のターゲットレートとされ、各GOPはこのターゲットレートに合わせてエンコードされる。

【0106】そして、ステップS36で、エンコード結果のビットストリームをRAID16に書き込む際のアドレスが設定され、ステップS37で、エンコーダ用コントロールファイルが出力される。このようにして作成されたコントロールファイルを用いることにより、ピクチャタイプが指定された固定レートの1バスエンコードが実行される。

【0107】ところが、上述した固定レートの1バスのエンコードでは、2バスエンコードに比べて作業時間を短縮できるものの、例えば複数のエンコード単位からなるビデオ素材のうちの任意のエンコード単位のビットレートだけを他のエンコード単位よりも高くすることなどはできなかった。このため、ビデオ素材のうちの、符号化が特に難しいと予測される部分についてのみビットレートを上げたいという要求や、エンコード結果の画質に問題があとときに、特定の範囲のビットレートだけを調整したいという要求には応じられないという問題がある。

【0108】以下では、上記の問題を解決するために提案する本発明の画像符号化方法について説明する。なお、ここでは、上述した2バスおよび固定レートの1バスによる画像符号化方法と共通する部分についての説明は省略し、これらとは異なる部分について主に説明するものとする。

【0109】図12は、本発明の画像符号化方法に係る1バスエンコードの処理手順を示すフローチャートである。この図12に示す処理手順において、ステップS42、ステップS44およびステップS46はオンラインで行われる処理であり、その他の各ステップの処理はオフラインで行われる。

【0110】まず、ステップS41で、エンコード条件が設定される。このエンコード条件は、従来と同様にスーパーバイザコントローラ1から送られる「v.enc」というファイルによって指定される。

【0111】次に、ステップS42で、カスタマイズ処理を行うためのカスタマイズパラメータが設定される。このカスタマイズパラメータは、前述したように、ビデオ素材に対する符号化ビットレートを配分する際の重み係数を含むものである。このカスタマイズパラメータにより、画質に問題があるとされた部分に対するビット配分を多く(ビットレートを高く)する、画質調整のためのカスタマイズが行われる。

【0112】このとき、ユーザーが、ビデオ素材に対して符号化レート配分などを変えて画質調整したい範囲を、エンコードの開始前に知っている場合には、最初に画質調整用のカスタマイズファイルが作成される。

【0113】ステップS43では、GOP毎のレート配分が計算され、1バスエンコード用のコントロールファイルが作成される。そして、ピクチャタイプごとに各GOP毎のターゲットレートが指定される。ここで、従来の1バスエンコードと異なる点は、GOP毎に割り当てられるビットレートが固定ではなく、異なるレートが指定されることである。このGOP毎のターゲットレートは、ビデオ素材の全再生時間が、記録媒体の記録容量や伝送路の伝送容量など、予め設定される総ビット量以内に収まるように決定される。

【0114】ステップS44では、ステップS43で作成されたコントロールファイルに従ってエンコードを実行するかどうかが判断される。実際には、ユーザーが、MPEGエンコーダ25に内蔵されているローカルデコーダから出力される画質をチェックするプレビュー(Preview)を行って判断する。

【0115】このプレビューは、上記のローカルデコーダから出力されるエンコード結果のビットストリームをRAID16に出力しないモードであり、ユーザがローカルデコードする範囲を任意に指定することができる。

【0116】そして、ステップS45で画質に問題がないと判断された場合には、ステップS46に進み、ステップS43で作成されたコントロールファイルに従ってエンコードを実行する。一方、ステップS45で画質に問題があると判断された場合には、ステップS42に戻って画質調整のためのカスタマイズ作業を行い、各GOPに対するレート配分計算を再度実行する。このカスタマイズ作業では、画質に問題がある部分のビットレートを高くしたり、フィルターレベルを調整することなどが行われる。

【0117】その後、ステップS46に進み、ユーザが、カスタマイズした部分の画質を確認し、全ての部分が良好であるとされれば全体のエンコードを実行させ

る。

【0118】そして、ステップS47では、エンコード結果であるビットストリームが、SCSIを経由して、オーサリングシステムのサーバーであるRAID16に直接書き込まれるなどの後処理が行われる。また、エンコード後には、エンコード結果情報が、ネットワーク経由でスーパーバイザーに報告される。

【0119】以上の手順は、本発明に係る画像符号化方法によるビデオ素材のエンコード処理を示しているが、プレビューによる画質評価を行う必要がない場合には、処理を簡略化することができる。

【0120】図13は、図12に示した処理手順を簡略化した手順を示している。すなわち、GOP毎のターゲットレートを部分的に変更する必要がないことが予め分かっている場合には、ステップS43でGOP毎のレート配分を計算した後に、プレビューによる画質確認を省略してステップS46に進み、エンコード素材全体のエンコード処理を実行するようにする。

【0121】このようにすれば、エンコード作業時間をさらに短縮することができる。

【0122】図14は、上述した本発明に係るエンコード処理手順のステップS43におけるGOPレート配分計算の手順を具体的に示している。

【0123】まず、ステップS51では、エンコード条件が入力される。ここで、ディスク容量の中からビデオ
weight.txt

に割り当てられたビット総量「QTY_BYTES」と、最大ビットレート「MAXRATE」に対して、最大ビットレート以下になるように制限を加えた総ビット量「USB_BYTES」を求め、この値からGOPヘッダに必要なビット量「TOTAL_HEADER」を差し引いた値と全体のフレーム総数からターゲット数の総和の目標値となる「TOTAL_SUPPLY」を算出する。

【0124】ステップS52では、カスタマイズファイルが読み込まれ、カスタマイズパラメータが設定される。

【0125】表1は、カスタマイズファイル「weight.txt」の例である。「ENCU_nb」はエンコードするロールの番号、「Filter」はフィルターの強さ、「D_weight」はビット配分の際の重み係数を意味している。また、「mquant_level」は、各ピクチャでの平坦な画像の場合のフレーム内の処理モードの選択「MQUANT ON/OFF」である。「scene_add」は、シーンチェンジ点などでPピクチャをIピクチャにマニュアルで変更したい場合に用いられる。ここでは、パラメータの変化点の時間情報がタイムコードとして表されている。なお、表1中では、「Filter」を「Fil」、「D_weight」を「D_w」、「scene_add」を「S_add」、「mquant_level」を「Mq」とそれぞれ略記している。

【0126】

【表1】

Timecode	ENCU_nb	Fil	D_w	S_add	Mq	pn_c	Re
01:00:00:00	1	2	10	0	1	0	0
01:00:10:14	1	2	20	0	1	0	0
01:00:22:05	1	2	10	0	1	0	0
01:01:00:00	2	2	10	0	1	0	0
01:01:20:29	2	0	10	0	1	0	0
01:01:28:26	2	2	10	0	1	0	0

【0127】カスタマイズパラメータの指定方法、およびカスタマイズファイルの作成手順については後述する。

【0128】ステップS53では、ピクチャタイプが指定される。これは、従来の場合と同じである。ここで、シーンチェンジが指定された（すなわち「weight.txt」の「scene_add」が1とされた）フレームでは、PピクチャがIピクチャに変換される。このとき、指定されたフレームがBピクチャである場合には、そのフレームから最も近いPピクチャがIピクチャに変換される。なお、このピクチャタイプの指定については、前述した場合と同様であるので、ここでは詳しい説明を省略する。

【0129】ステップS54ではシーンチェンジ処理が

$$\text{difficulty}[k] = \text{d_weight}[k] * KX * \text{genbit}$$

行われ、ステップS55ではチャプタ境界処理が行われる。このシーンチェンジ処理およびチャプタ境界処理についても、前述した場合と同様であるので、ここでは詳しい説明を省略する。

【0130】そして、ステップS56で、上記のように指定されたピクチャタイプに対して、仮想的な符号化難易度（Difficulty）の値が作成される。

【0131】次に、この仮想符号化難易度の値を作成する方法について説明する。

【0132】仮想符号化難易度は（18）式で表される。

【0133】

ここで、「d_weight[k]」はエンコードユニットENCUの先頭からk番目のフレームの重み係数の値、「const」は定数、「KX」はピクチャタイプに対する重み係数であり、「genbit」は各ピクチャタイプ(I, P, Bビ

$$KX = Ki, \text{genbit} = Gi$$

p_type[k] = "P" (Pピクチャ) のとき

$$KX = Kp, \text{genbit} = Gp$$

p_type[k] = "B" (Bピクチャ) のとき

$$KX = Kb, \text{genbit} = Gb$$

これらの値は、例えば、 $Ki = 1.0$, $Gi = 40000$, $Kp = 1.0$, $Gp = 20000$, $Kb = 1.4$, $Gb = 10000$ などである。そして、「difficulty[k]」には、「weight.txt」に設定された重み付けが反映される。

【0135】次に、各ENCUのターゲット数の総和の

$$Qt[1] = \text{frame_number}[1] \times e_weight[1] \quad (22)$$

$$Qt[2] = \text{frame_number}[2] \times e_weight[2] \quad (23)$$

$$\text{SUPPLY_BYTES}[1] = \text{TOTAL_SUPPLY} \times Qt[1] / (Qt[1] + Qt[2]) \quad (24)$$

$$\text{SUPPLY_BYTES}[2] = \text{TOTAL_SUPPLY} \times Qt[2] / (Qt[1] + Qt[2]) \quad (25)$$

ここで、「frame_number[1]」, 「frame_number[2]」は「ENCU 1」, 「ENCU2」のフレーム数であり、「e_weight[1]」, 「e_weight[2]」は各ENCUのレート配分時に対する重み係数である。

【0138】そして、ステップS57では、GOPレートが計算される。ここでは、各GOPに対して、仮想的

$$\text{DIFFICULTY_SUM} = \sum (\text{difficulty}[k]) \quad (26)$$

$$B = \text{GOP_MINBYTES} \quad (27)$$

$$\Sigma y = A \times \Sigma x + B \times n \quad (28)$$

ここで、 $\Sigma y = \text{SUPPLY_BYTES}[1]$, $\Sigma x = \text{DIFFICULTY_SUM}$ であり、nはGOPの総数である。よって、(28)式が得られ、「gop_target[j]」および「target_r

$$A = (\text{SUPPLY_BYTES}[1] - B \times n) / \text{DIFFICULTY_SUM} \quad (29)$$

$$\text{gop_target}[j] = A \times (\text{gop_diff}[j]) + B \quad (30)$$

$$\text{target_rate}[j] = \text{gop_target}[j] / \text{gop_nb}[j] / KT \quad (31)$$

jはENCUの最初から数えたGOPの番号、「gop_diff」および「gop_nb」はj番目のGOPに含まれるピクチャの符号化難易度の総和およびピクチャの数である。

また、「GOP_MINBYTES」はGOP単位での最小ターゲットバイト数(定数)である。各GOPは、このターゲットレートに合わせてエンコード処理されることになる。

【0140】ステップS58では、エンコード結果のビットストリームが書き込まれるRAID16のアドレスが計算される。

【0141】そして、ステップS59で、エンコーダ用コントロールファイルが出力され、処理が終了する。図16は、カスタマイズパラメータの指定方法の具体例を示している。

【0142】「WINDOW(1)」は、ビット計算の結果の各ピクチャに対するターゲットビット量を棒グラフで示

クチャ)に対する仮想的な符号化難易度である。この「KX」と「genbit」は、(19)～(21)式で与えられる。

$$[0134] p_type[k] = "I" \quad (I\text{ピクチャ}) \text{のとき} \quad (19)$$

$$(20)$$

$$(21)$$

目標値「SUPPLY_BYTEx」を求める。

【0136】図15は、総ビット量と「SUPPLY_BYTEx」との関係を示している。例えば、ENCUが2つの場合には次の式で求められる。

$$[0137]$$

$$(22)$$

$$(23)$$

$$(24)$$

$$(25)$$

な符号化難易度(difculty)を用い、図7のような評価関数を用いて符号化ビットレートの配分が計算される。「ENCU 1」の各GOPへのレート配分は、次の(26)式～(31)式により求められる。「ENCU 2」に対するレート配分についても同様である。

$$(26)$$

$$(27)$$

$$(28)$$

ate[j]」は、(30)式および(31)式で与えられる。

$$[0139]$$

$$(29)$$

$$(30)$$

$$(31)$$

しており、縦方向はビットの大きさ、横方向は時間軸である。

【0143】ユーザーは、エンコードした結果を見て、画質を改善するためにターゲットビット量を調整したい部分の仮想的な符号化難易度の重み係数「d_weight」を指定する。また、「WINDOW(2)」は、各ピクチャに対する仮想的な符号化難易度の重み係数「d_weight」を示している。なお、「WINDOW(3)」は、「cs enable [k]」により再配分される0の範囲を示している。

【0144】この例では、各ピクチャに対する符号化ビットレートは、重み付け後の符号化難易度に応じて配分されているため、「WINDOW(2)」で重み係数を大きくすると、レート配分計算の結果、その分だけレートが増加することになる。

【0145】ユーザーは、図中のTSからTEまでの力

スタマイズ範囲と、重み係数の大きさを指定する。ここで、TSは「START POINT」を示しており、TEは「END POINT」を示している。フィルターなどの他のパラメータも同様に指定される。

【0146】このようなカスタマイズパラメータ指定作業の結果として、カスタマイズ用のコントロールファイル「weight.txt」が作成される。

【0147】図17は、上述したカスタマイズファイルの作成手順を示すフローチャートである。ステップS61では、VBV制限を行うスタート点「kstart」およびエンド点「kend」が設定され、カスタマイズ範囲が入力される。

【0148】ステップS62では、カスタマイズレベルが指定される。このカスタマイズレベルは、画質を改善

Timecode 指定する Customize menu

するためにターゲットビット量を調整したい部分の仮想的な符号化難易度の重み係数「d_weight」により指定される。

【0149】そして、ステップS63で、カスタマイズ用のコントロールファイル「weight.txt」が作成される。なお、ここで作成されるカスタマイズ用のコントロールファイル「weight.txt」は、前述の表1に示したものと同様である。

【0150】表2は、タイムコード指定するためのカスタマイズメニューの例を示している。これにより仮想的な符号化難易度の重みを変えて、特定の範囲のターゲットビット量を調整する。

【0151】

【表2】

diff_weight	2 -- 30)	/* default 10 */
pre filter level	0 -- 7)	/* default filter_level */
nquant level	0 -- 1)	/* default 1 */
scene_add	0 -- 1)	/* default 0 */

(「1」でPピクチャをIピクチャに変更する。)

【0152】ところで、ユーザーが、エンコード素材の段階でENCU間のビット配分を操作したい場合がある。例えば、映画素材の前に挿入される宣伝用の画像だけはビットレートを上げたいという場合である。このような場合には、各ENCUのレート配分時に、重み係数「e_weight」を設定する。この重み係数は、例えば、表3のようなファイルにより設定され、レート配分計算時に読み込まれる。

【0153】

【表3】

encu_weight.txt : default 100

ENCU_nb	e_weight
1	120
2	100

【0154】なお、ここでは、ENCUが2個の場合について説明したが、ENCUが2個より多い場合にも同様に計算できる。このようにして作成されたコントロールファイルによるエンコード処理を行うことにより、カスタマイズパラメータが反映された1バスエンコードが実現される。

【0155】次に、本発明に係る画像符号化方法において、エンコード素材が映画フィルムに記録された映像である場合の処理手順について説明する。

【0156】映画フィルムに記録された映像を、ビデオ映像規格のメディアに記録するためには、オーサリングに先だって、デジタルVTRに記録してマスタを作成するテレシネ変換と呼ばれる作業が行われる。このテレシネ変換の際には、24コマ/秒で構成されているフィルム映像を、例えばNTSC方式の30フレーム/秒に変換するプルダウン(pull down)と呼ばれる処理も行われる。この処理は、同じフィールドを周期的に繰り返すものであり、2-3プルダウンなどと呼ばれる。

【0157】本発明に係るエンコードシステムでは、圧縮符号化されたビデオ素材をDVDに蓄積する際には24コマ/秒のまでディスクに記録し、再生時にDVDプレーヤ内部の信号処理により、これを30フレーム/秒に変換することにより記録されるデータ量を低減している。

【0158】図18は、このような2-3プルダウンを含むエンコードの処理手順を示すフローチャートである。

【0159】ステップS91では、まずエンコード条件が入力される。

【0160】そして、ステップS92で、上記の入力されたエンコード条件に基づいて、プルダウンパターンが測定される。なお、この測定は、プルダウン指定時にのみ各ENCUごとに30秒間ずつ行われる。

【0161】ステップS93では、上記のプルダウンパ

ターンに従ってレート配分計算が行われる。

【0162】そして、ステップS94で、エンコードを実行するかどうかが判断される。ここで、上記の計算されたレート配分に基づいてエンコードを実行すると画質に問題があると判断された場合にはステップS95に進み、そのままエンコードを実行してもよいと判断された場合にはステップS98に進む。

【0163】ステップS95では、前述した手順と同様にカスタマイズ条件が設定される。

【0164】ステップS96では、上記の設定されたカスタマイズ条件に基づいてレートの再配分計算が行われる。そして、必要に応じてステップS97でプレビューを行って、ユーザが画質を確認した後に、再びステップS94に戻り、エンコードを実行するかどうかが判断される。

【0165】一方、ステップS94でエンコードを実行してもよいと判断された場合には、ステップS98でそのままエンコードが実行される。

【0166】そして、ステップS99で、エンコードを終了するかどうかが判断される。ここで、エンコードを終了してもよいと判断された場合にはステップS100に進み、エンコードをやり直す必要があると判断された場合にはステップS95以降のステップを繰り返す。

【0167】そして、ステップS100で、エンコード結果のビットストリームをRAID16に書き込むためのアドレスの設定や、エンコーダ用コントロールファイルを出力する等の後処理が行われてエンコード作業を終了する。

【0168】ところで、以上の本発明に係る画像符号化方法についての説明は、ビデオ素材を圧縮符号化してパッケージメディアに蓄積することを前提としている。しかし、本発明の画像符号化方法により、GOP毎に符号化ビットレートが配分されて圧縮符号化されたビデオ素材は、パッケージメディアに蓄積されるのみでなく、エンコードシステムから他の機器に伝送されることもある。

【0169】すなわち、本発明の画像伝送方法は、ビデオ素材に対してビット配分の重み係数を含むカスタマイズパラメータを設定した後に、そのカスタマイズパラメータに基づいて、GOPを単位とする符号化ビットレートを、それらの総データ量が予め設定された所定の伝送データ量を越えないように配分するものである。

【0170】この画像伝送方法によれば、前述した本発明に係る画像符号化装置を用いて、短時間にエンコード作業時間を行うことができ、ビデオ素材の任意の部分に符号化ビットを配分できる画像伝送方法を提供できる。

【0171】

【発明の効果】本発明によれば、固定レートの1バスエンコードにおいて、仮想的な符号化難易度を導入してビット配分を行うようにしたため、エンコード作業時間を

短縮すると共に、ユーザーの多様な要求に応じて任意に符号化ビットレートを配分できる画像符号化方法および装置、画像伝送方法を提供できる。

【図面の簡単な説明】

【図1】本発明に係る画像符号化装置の構成例を示すブロック図である。

【図2】GOPについて説明するための図である。

【図3】2バスエンコードの手順を示すフローチャートである。

【図4】2バスエンコードにおけるコントロールファイルの作成手順を示すフローチャートである。

【図5】シーンチェンジについて説明するための図である。

【図6】チャプタ処理について説明するための図である。

【図7】「GOP_DIFF」と「GOP_TARGET」量とを変換するための関数の一例を示す図である。

【図8】VBVの計算について模式的に示す図である。

【図9】GOP単位のターゲットビット配分計算の一例を示す図である。

【図10】1バス固定レートエンコードの手順を示すフローチャートである。

【図11】1バス固定レートエンコードにおけるコントロールファイルの作成手順を示すフローチャートである。

【図12】本発明に係る画像符号化方法の処理手順を示すフローチャートである。

【図13】本発明に係る画像符号化方法の別の処理手順を示すフローチャートである。

【図14】本発明に係る画像符号化方法におけるコントロールファイルの作成手順を示すフローチャートである。

【図15】総ビット量と「SUPPLY_BYTES」との関係を示す図である。

【図16】カスタマイズパラメータの指定方法の様子を説明するための図である。

【図17】本発明に係る画像符号化方法におけるカスタマイズファイルの作成手順を示すフローチャートである。

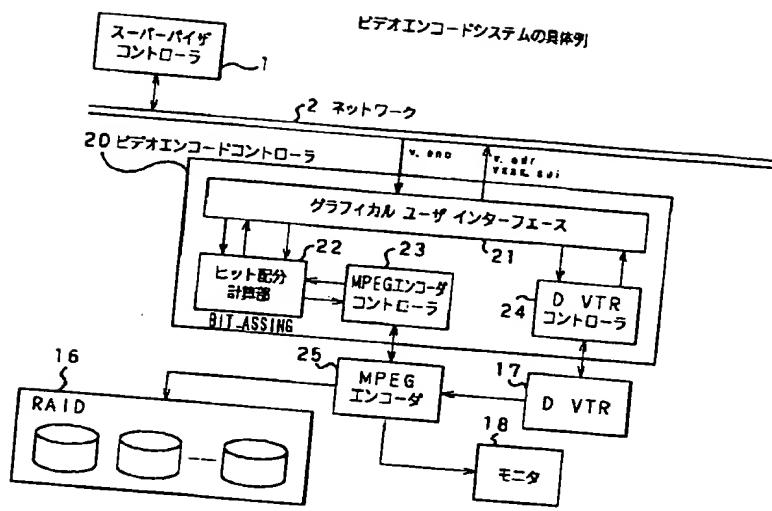
【図18】本発明に係る画像符号化方法を映画素材のフルダウントリミングに適用した場合の処理手順を示すフローチャートである。

【符号の説明】

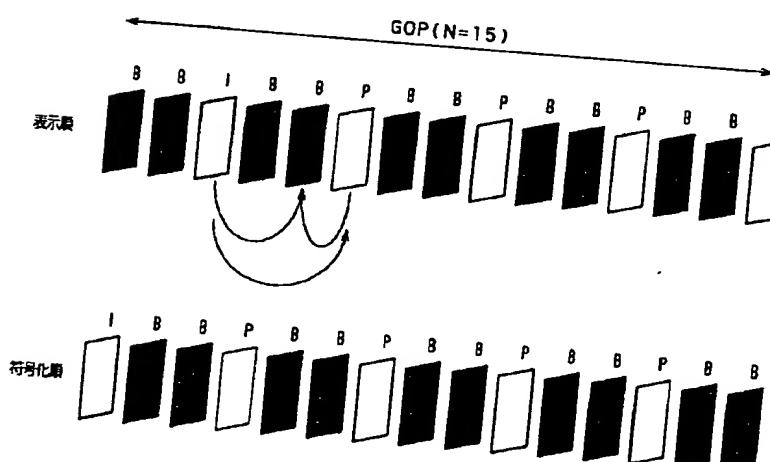
1 スーパーバイザコントローラ、 2 ネットワーク、 16 RAID、 17 DVTR (デジタルVTR)、 18 モニタ、 20 ビデオエンコードコントローラ、 21 グラフィカルユーザインターフェース (GUI)、 22 ビット配分計算部、 23 MPEGエンコーダコントローラ、 24 デジタルVTR (DVTR) コントローラ、 25 MPEG工

シコーダ

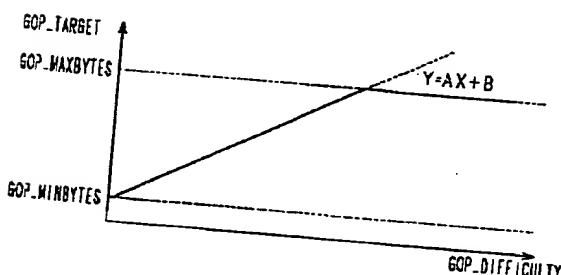
【図1】



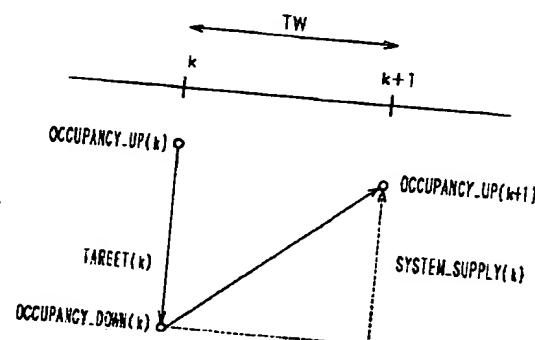
【図2】



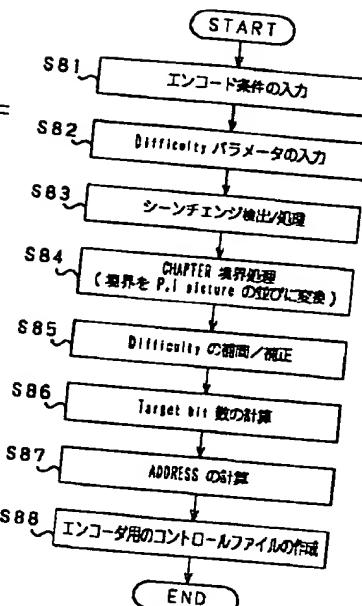
【図7】



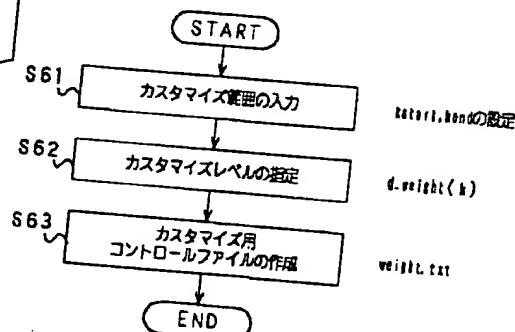
【図8】



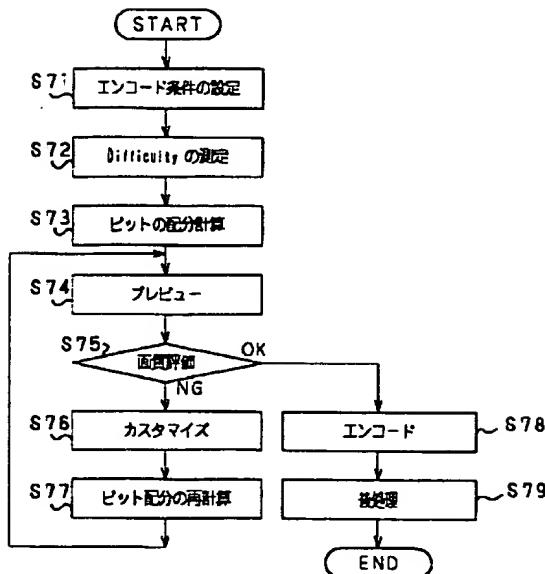
【図4】



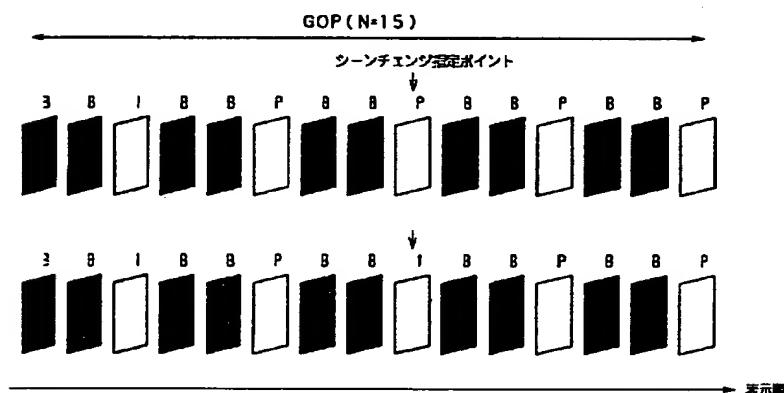
【図17】



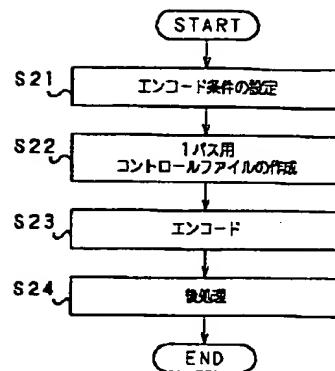
【図3】



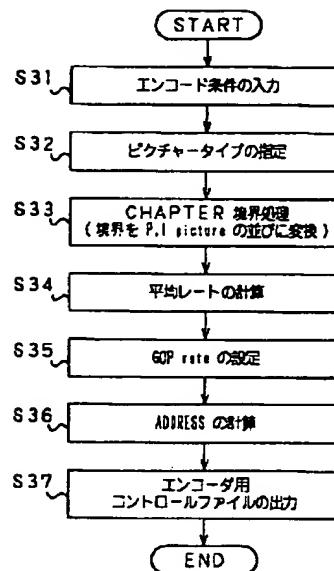
【図5】



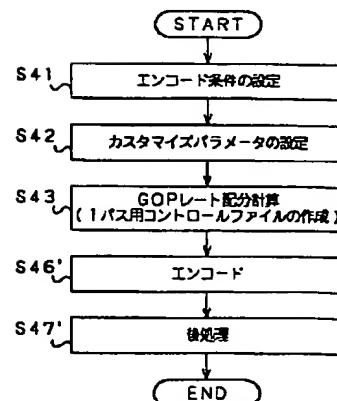
【図10】



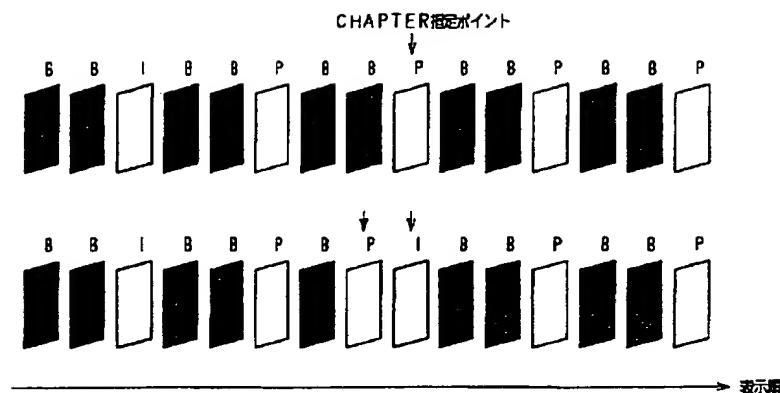
【図11】



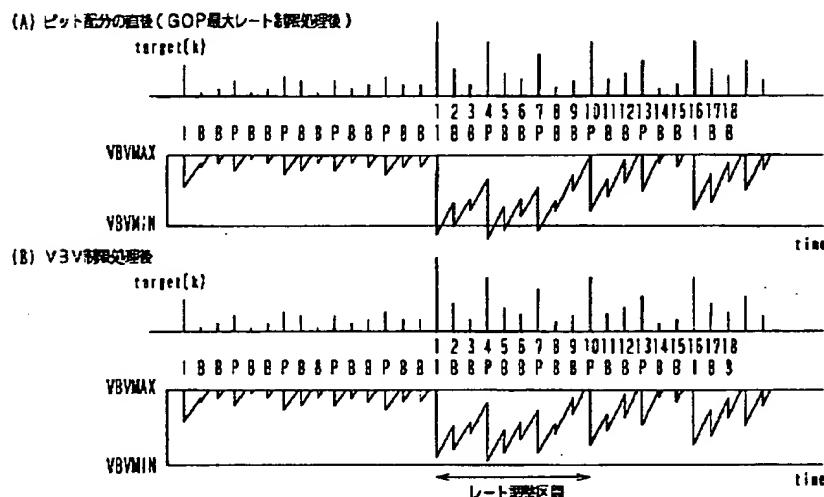
【図13】



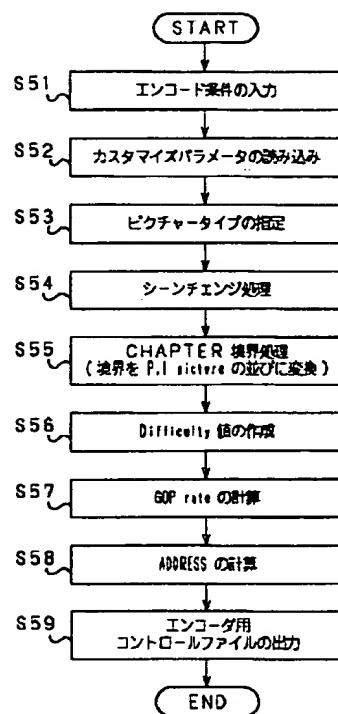
[図 6]



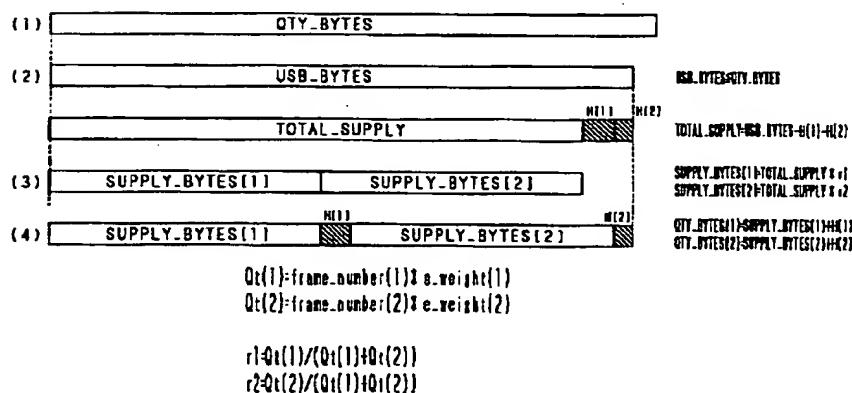
[9]



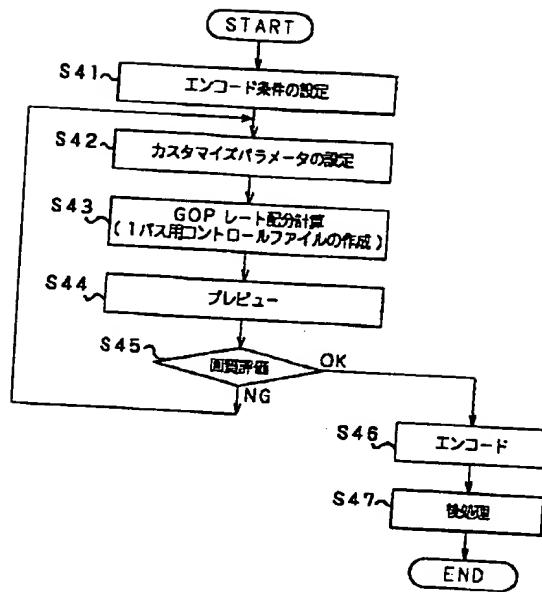
[図14]



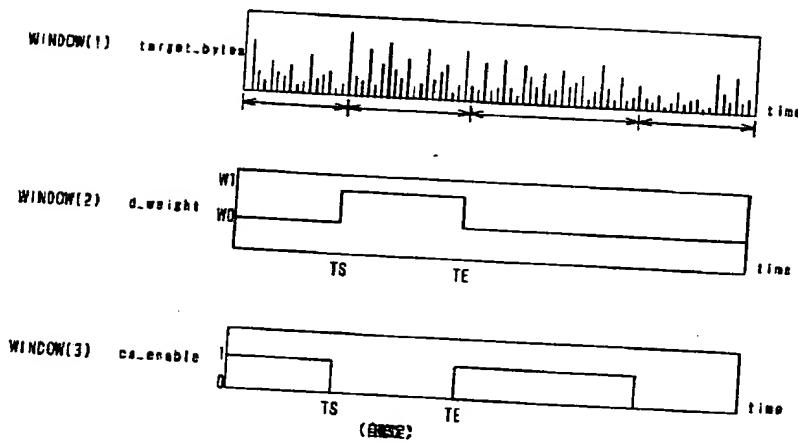
[图 15]



【図12】



【図16】



【図18】

